MATHÉMATIQUES ET INFORMATIQUE
Sciences
Université de Paris

UNIVERSITE DE PARIS
UFR MATHEMATIQUES ET INFORMATIQUE

Rapport sur les Ateliers Pratiques Google Cloud

Master 1 Vision Machine Intelligente

Merve ERBAS

Encadré par Benoit CHARROUX

Année universitaire 2023 – 2024

Table Des Matières

1. Introduction

Ce rapport présente une révision des ateliers pratiques que j'ai suivis sur Google Cloud. Ces ateliers m'ont permis d'acquérir des compétences pratiques en utilisant divers services et outils de Google Cloud Platform (GCP). Chaque étape de l'atelier sera détaillée avec les objectifs, les actions entreprises et les résultats obtenus.

2. Objectifs des Ateliers

Les principaux objectifs des ateliers étaient :

- Comprendre les principes de base de Google Cloud Platform.
- Apprendre à déployer des applications sur GCP.
- Utiliser des services spécifiques de GCP.
- Apprendre à configurer et à sécuriser les services cloud.
- Analyser des données en utilisant les outils de Google Cloud.

Les ateliers suivis incluaient :

1. Introduction à Google Cloud Platform

2. Développement d'applications : déployer l'application dans Kubernetes Engine – Python

3. Infrastructure as Code avec Terraform

3. Étapes et Résultats

Étape 1 : Présentation des ateliers pratiques Google Cloud

Se familiariser avec l'interface de Google Cloud Console et comprendre les concepts de base.

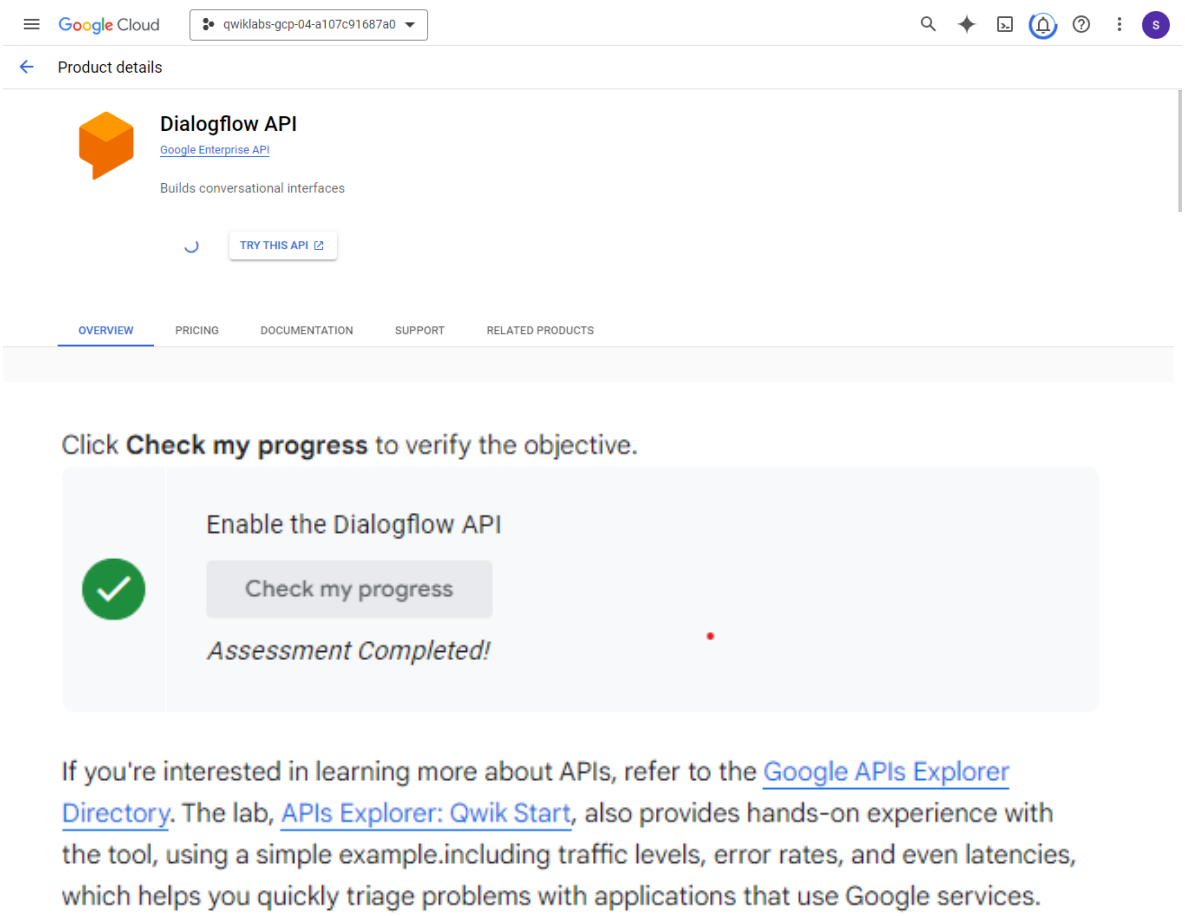
Actions :

- Accéder à la console Cloud

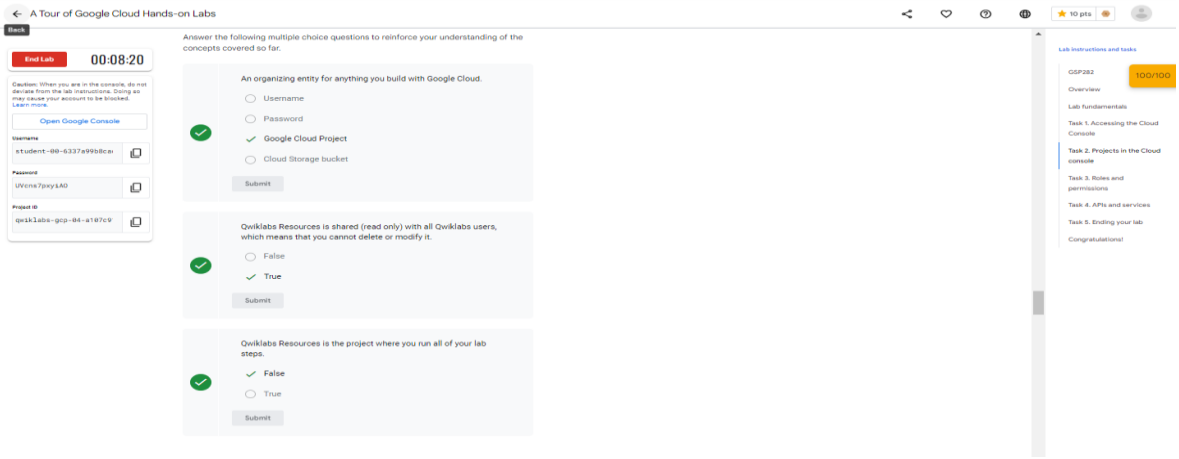
- Les projets dans la console Cloud

- Rôles et autorisations.

-API et services



-Terminer l’atelier



Étape 2 : Développement d'applications : déployer l'application dans Kubernetes Engine – Python

Google Kubernetes Engine (GKE) est un service géré qui permet de déployer, gérer et faire évoluer des applications conteneurisées à l'aide de l'infrastructure Google. GKE utilise des clusters de machines virtuelles pour exécuter des conteneurs orchestrés par Kubernetes, offrant des fonctionnalités telles que la gestion automatique des pods, la mise à l'échelle automatique et la mise à jour continue.

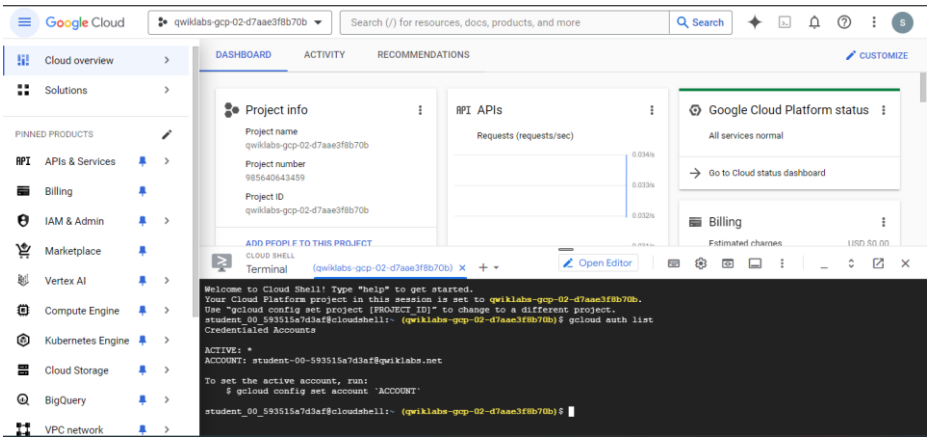
Dans cet atelier, nous avons déployé l'application Quiz dans Kubernetes Engine. Nous avons utilisé les services de Google Cloud Platform, tels que Container Builder (maintenant Cloud Build) et Container Registry, ainsi que les ressources Kubernetes comme les déploiements, les pods et les services.

Objectifs :

- Les objectifs de cet atelier étaient les suivants :
- Créer des Dockerfiles pour le frontend et le backend de l'application Quiz.
- Utiliser Cloud Build pour générer des images Docker.
- Provisionner un cluster GKE pour héberger l'application Quiz.
- Utiliser des déploiements Kubernetes pour provisionner des pods répliqués.
- Configurer un service Kubernetes pour créer un équilibreur de charge pour le frontend de l'application.

Actions :

1) **Activer Cloud Shell**



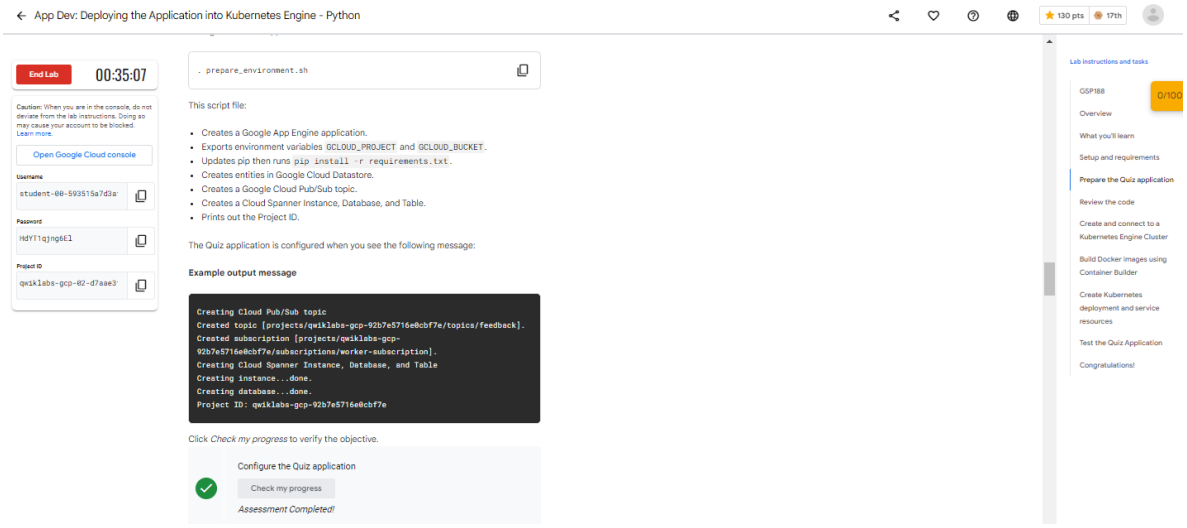
2)Préparer l'application Quiz

Modifiez le répertoire de travail :

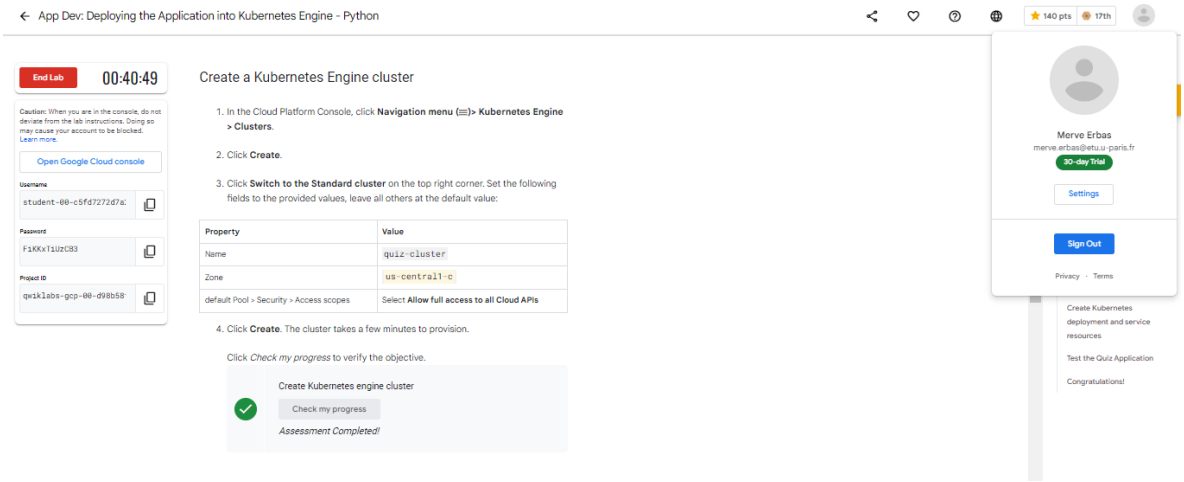
cd ~/kubernetesengine/start

. prepare_environment.sh

```
Downloading certifi-2024.2.2-py3-none-any.whl (163 kB)
     ---------------------------------------- 163.8/163.8 kB 15.3 MB/s eta 0:00:00
Downloading charset_normalizer-3.3.2-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (142 kB)
     ---------------------------------------- 142.1/142.1 kB 11.5 MB/s eta 0:00:00
Downloading googleapis_common_protos-1.63.0-py2.py3-none-any.whl (229 kB)
     ---------------------------------------- 229.1/229.1 kB 18.2 MB/s eta 0:00:00
Downloading idna-3.7-py3-none-any.whl (66 kB)
     ---------------------------------------- 66.8/66.8 kB 5.8 MB/s eta 0:00:00
Downloading MarkupSafe-2.1.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Downloading pyasn1-0.6.0-py2.py3-none-any.whl (85 kB)
     ---------------------------------------- 85.3/85.3 kB 4.4 MB/s eta 0:00:00
Installing collected packages: google-cloud, urllib3, sqlparse, pyasn1, protobuf, MarkupSafe, itsdangerous, idna, grpcio, google-crc3
c, click, charset-normalizer, certifi, cachetools, blinker, Werkzeug, rsa, requests, pyasn1-modules, proto-plus, Jinja2, grpcio-tools
 grpcio-gcp, grpc-interceptor, googleapis-common-protos, google-resumable-media, grpcio-status, google-auth, Flask, grpc-google-iam-v
, google-api-core, google-cloud-core, google-cloud-storage, google-cloud-spanner, google-cloud-pubsub, google-cloud-language, google-
loud-datastore
Successfully installed Flask-3.0.3 Jinja2-3.1.4 MarkupSafe-2.1.5 Werkzeug-3.0.3 blinker-1.8.2 cachetools-5.3.3 certifi-2024.2.2 chars
t-normalizer-3.3.2 click-8.1.7 google-api-core-2.19.0 google-auth-2.29.0 google-cloud-0.34.0 google-cloud-core-2.4.1 google-cloud-dat
store-2.19.0 google-cloud-language-2.13.3 google-cloud-pubsub-2.21.1 google-cloud-spanner-3.46.0 google-cloud-storage-2.16.0 google-c
c32c-1.5.0 google-resumable-media-2.7.0 googleapis-common-protos-1.63.0 grpc-google-iam-v1-0.13.0 grpc-interceptor-0.15.4 grpcio-1.63
0 grpcio-gcp-0.2.2 grpcio-status-1.62.2 grpcio-tools-1.62.2 idna-3.7 itsdangerous-2.2.0 proto-plus-1.23.0 protobuf-4.25.3 pyasn1-0.6.
 pyasn1-modules-0.4.0 requests-2.31.0 rsa-4.9 sqlparse-0.5.0 urllib3-1.26.18
Creating Datastore entities
Export credentials key.json
Creating Cloud Pub/Sub topic
Created topic [projects/qwiklabs-gcp-02-d7aae3f8b70b/topics/feedback].
Created subscription [projects/qwiklabs-gcp-02-d7aae3f8b70b/subscriptions/worker-subscription].
Creating Cloud Spanner Instance, Database, and Table
Creating instance...done.
Creating database...done.
Project ID: qwiklabs-gcp-02-d7aae3f8b70b
(developingapps) student_00_593515a7d3af@cloudshell:~/kubernetesengine/start (qwiklabs-gcp-02-d7aae3f8b70b)$
```



-Créer un cluster Kubernetes Engine et s'y connecter



-Connect to the cluster

kubectl get pods

```
or to unset it, run:

  $ gcloud config unset project
student_00_c5fd7272d7a2@cloudshell:~ (qwiklabs-gcp-00-d98b58fad461)$ gcloud container clusters get-credentials quiz-cluster --zone us-central1
-c --project qwiklabs-gcp-00-d98b58fad461
Fetching cluster endpoint and auth data.
kubeconfig entry generated for quiz-cluster.
student_00_c5fd7272d7a2@cloudshell:~ (qwiklabs-gcp-00-d98b58fad461)$ kubectl get pods
No resources found in default namespace.
student_00_c5fd7272d7a2@cloudshell:~ (qwiklabs-gcp-00-d98b58fad461)$
```

- Créer des images Docker avec Container Builder

Dans l'éditeur de code Cloud Shell, ouvrez `frontend/Dockerfile`. Vous allez ensuite ajouter un bloc de code qui exécute les actions suivantes :

- Saisie de la commande Dockerfile pour initialiser la création d'une image Docker personnalisée avec, pour point de départ, l'image Python App Engine de Google
- Codage des commandes Dockerfile pour activer un environnement virtuel
- Codage de la commande Dockerfile pour exécuter `pip install` dans le cadre du processus de création
- Codage de la commande Dockerfile pour ajouter les contenus du dossier actuel au chemin `/app` dans le conteneur
- Création du `Dockerfile` à l'aide de l'instruction `gunicorn…` qui s'exécute lorsque le conteneur est ouvert. Gunicorn (Green Unicorn) est un serveur HTTP compatible avec la spécification de l'interface passerelle de serveur Web (WSGI) Python.

On copie ensuite le code suivant et colles dans le fichier `Dockerfile` :

FROM gcr.io/google_appengine/python

RUN virtualenv -p python3.7 /env

ENV VIRTUAL_ENV /env

ENV PATH /env/bin:$PATH

ADD requirements.txt /app/requirements.txt

RUN pip install -r /app/requirements.txt

ADD . /app

CMD gunicorn -b 0.0.0.0:$PORT quiz:app

On ouvre ensuite le fichier `backend/Dockerfile`, puis copiez et collez le code suivant :

FROM gcr.io/google_appengine/python

RUN virtualenv -p python3.7 /env

ENV VIRTUAL_ENV /env

ENV PATH /env/bin:$PATH

ADD requirements.txt /app/requirements.txt

RUN pip install -r /app/requirements.txt

ADD . /app

CMD python -m quiz.console.worker

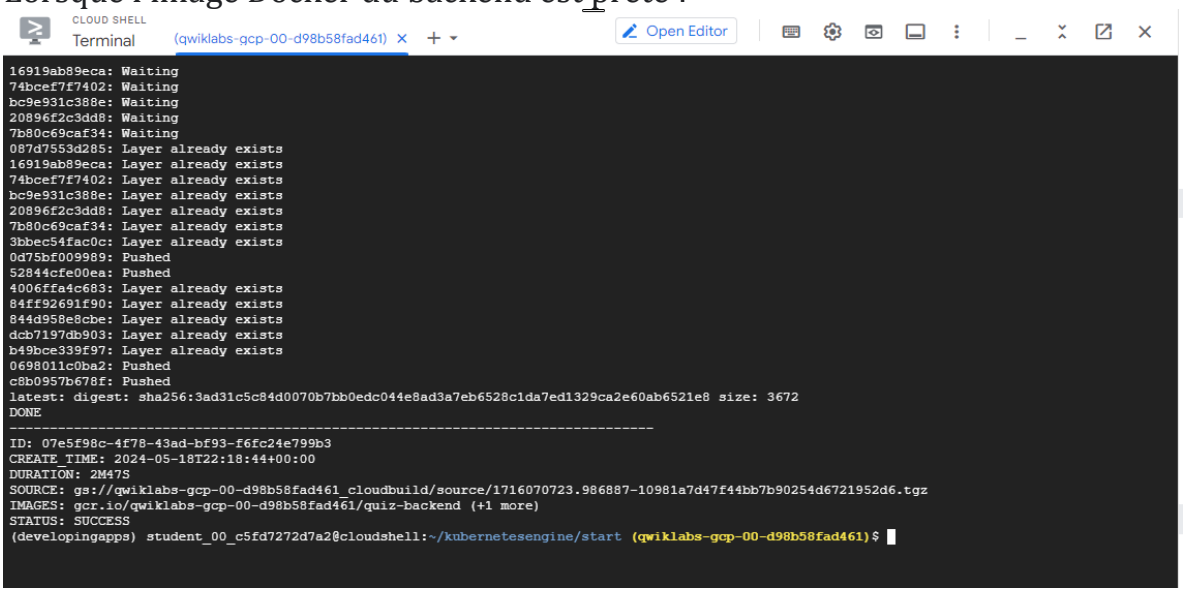
- On exécute la commande suivante pour créer l'image Docker frontale:

gcloud builds submit -t gcr.io/$DEVSHELL_PROJECT_ID/quiz-frontend ./frontend/

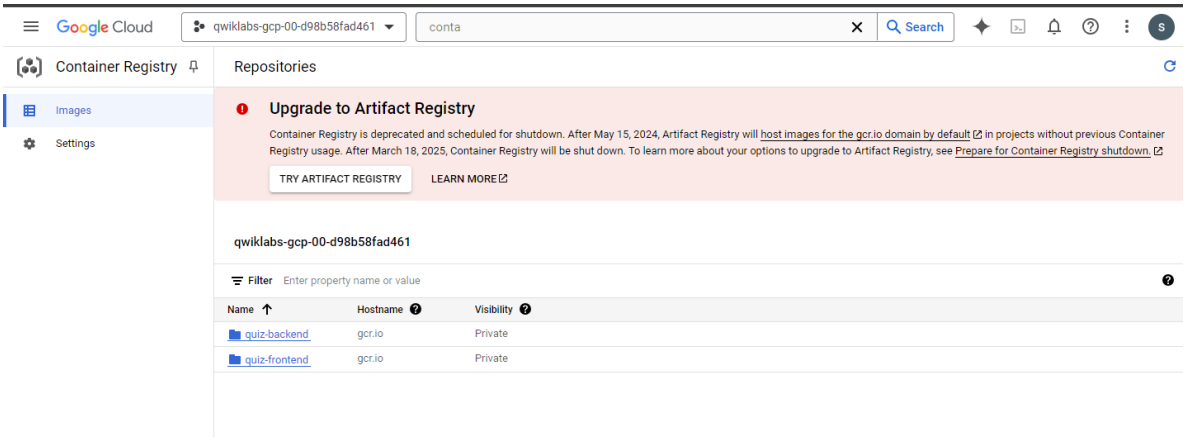
et pour créer l'image Docker du backend :

gcloud builds submit -t gcr.io/$DEVSHELL_PROJECT_ID/quiz-backend ./backend/

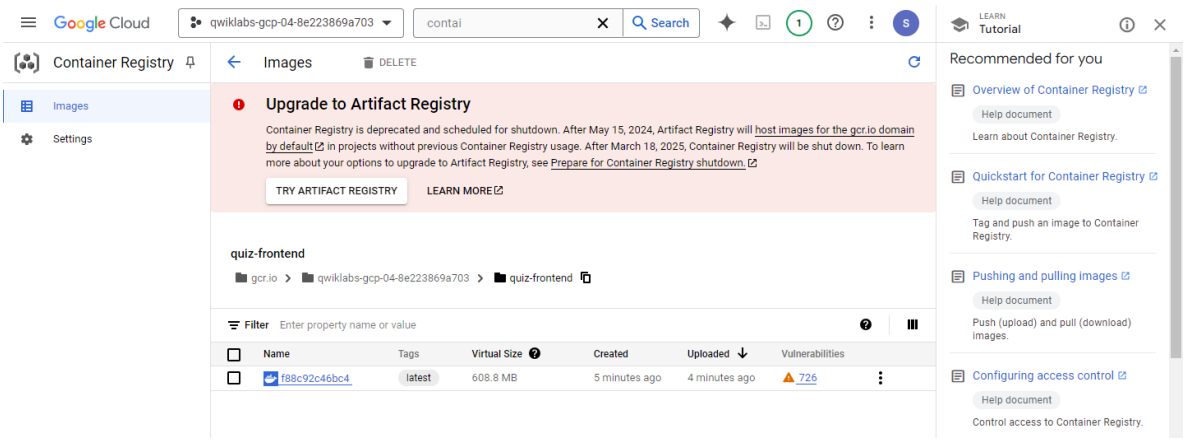
Lorsque l'image Docker du backend est prête :



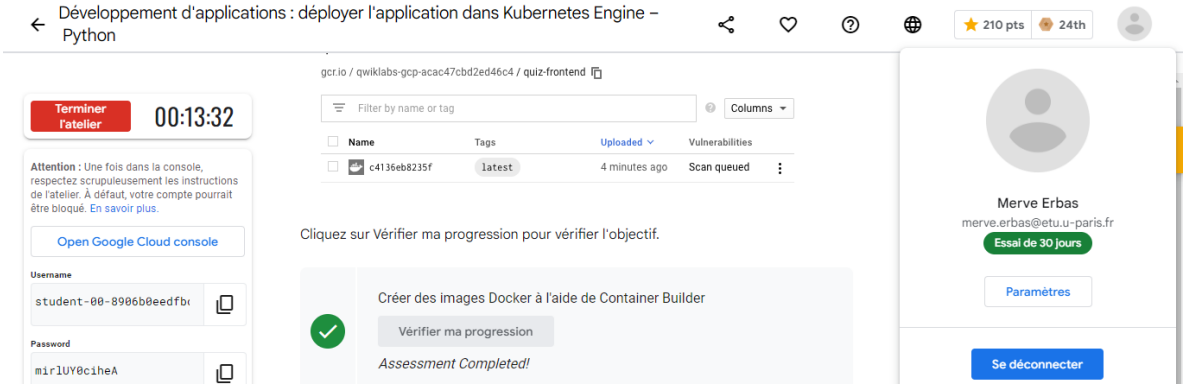
On peut voir les deux pods suivants : `quiz-frontend` et `quiz-backend` sur **Container Registry**.



On cliques sur ***quiz-frontend :***



Et j'ai complété cet étapes aussi :



- Créer des ressources de déploiement et de service Kubernetes

Dans Cloud Shell, provisionnez le déploiement de l'interface de Quiz.

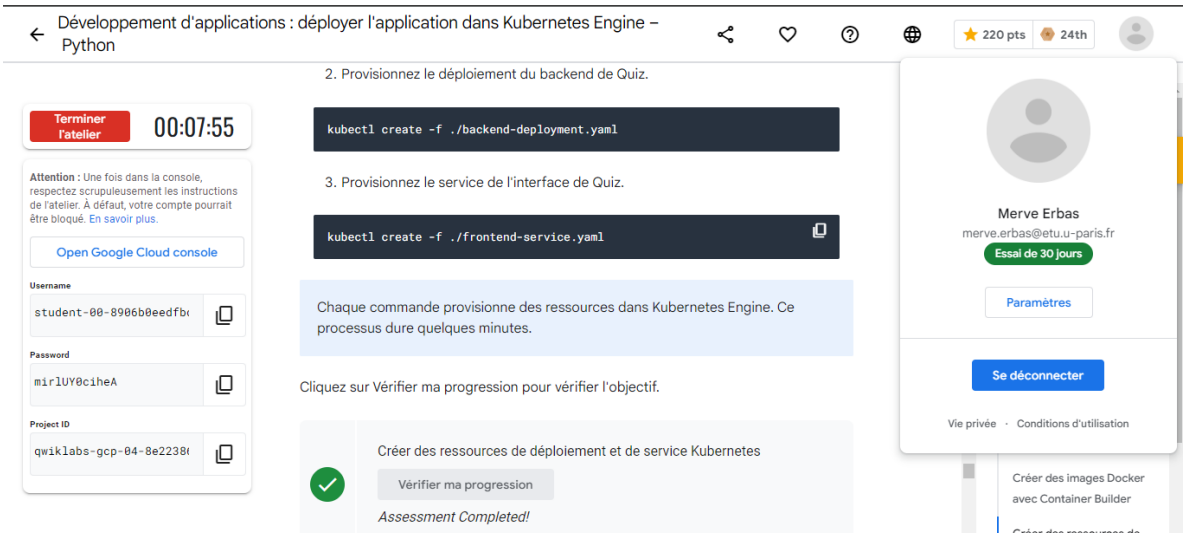
kubectl create -f ./frontend-deployment.yaml

Provisionnez le déploiement du backend de Quiz.

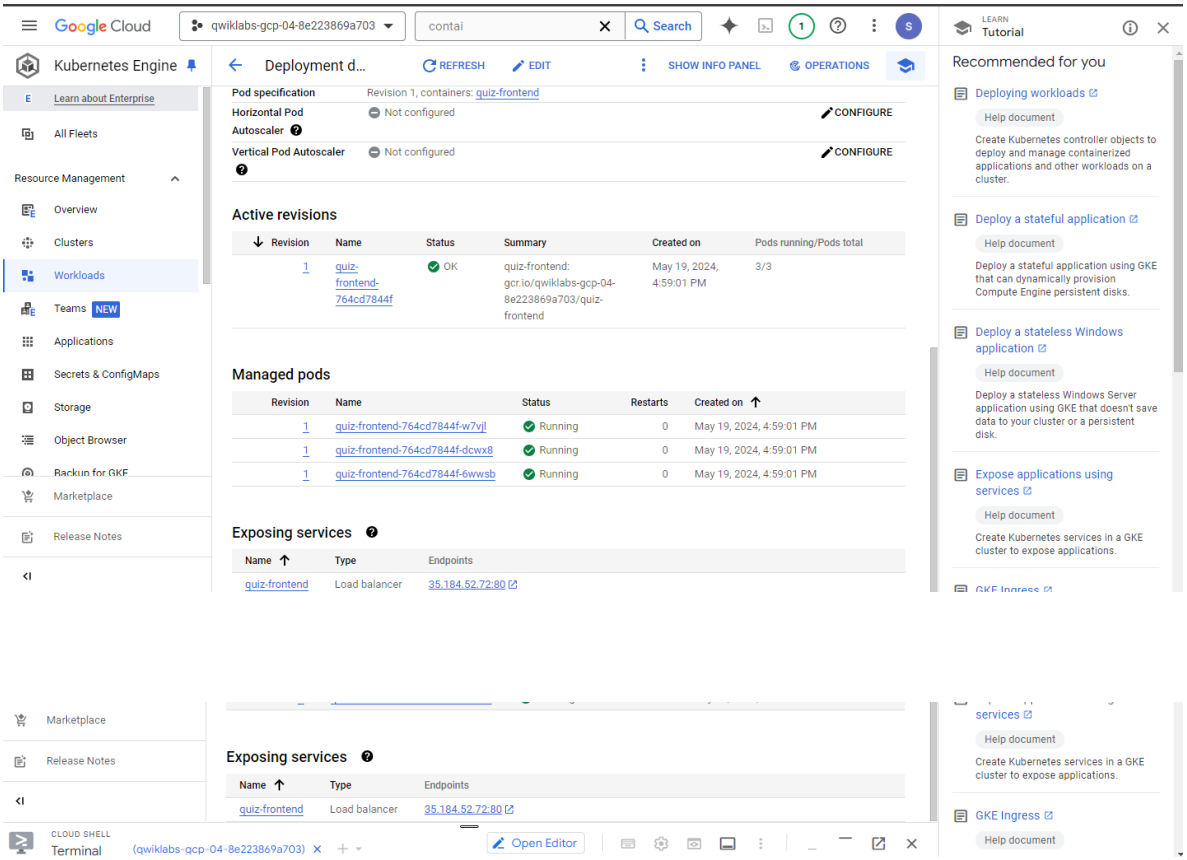
kubectl create -f ./backend-deployment.yaml

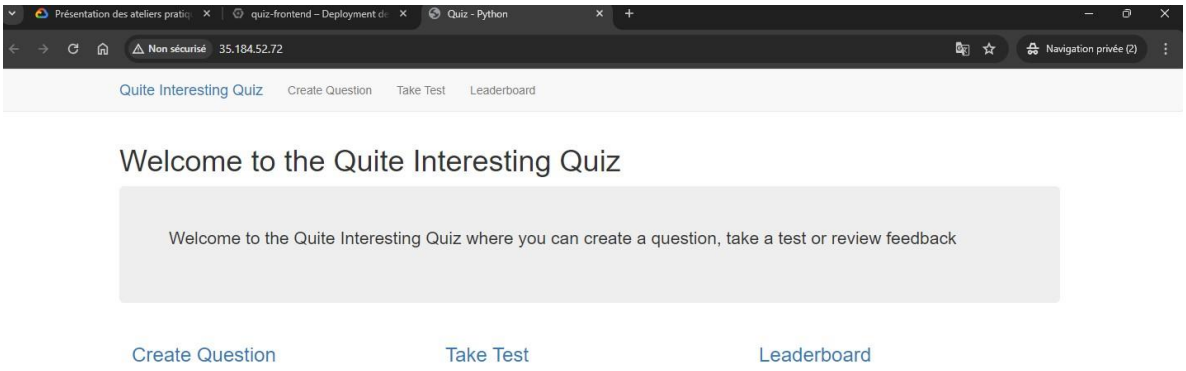
Provisionnez le service de l'interface de Quiz.

kubectl create -f ./frontend-service.yaml



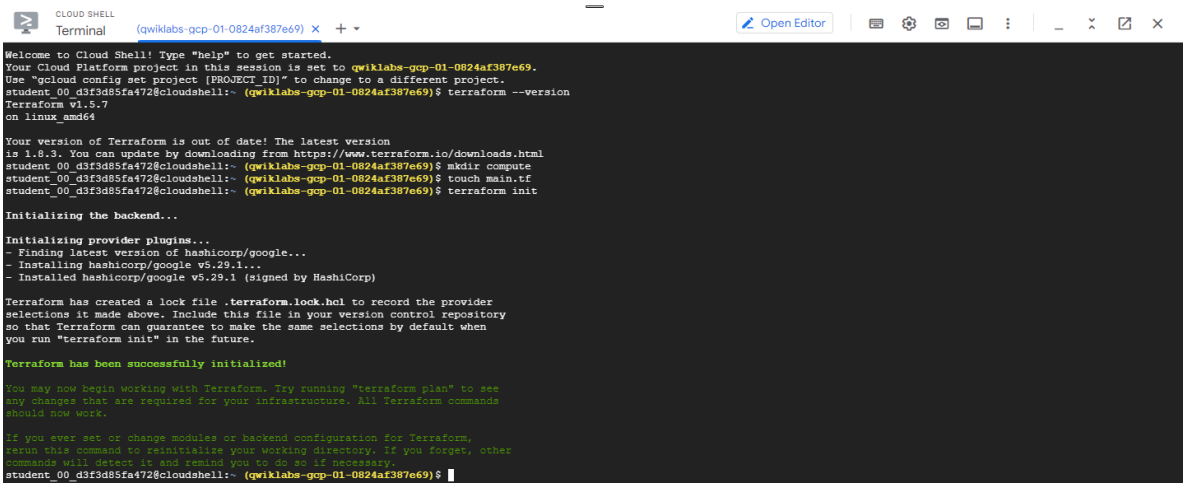
-Tester l'application Quiz



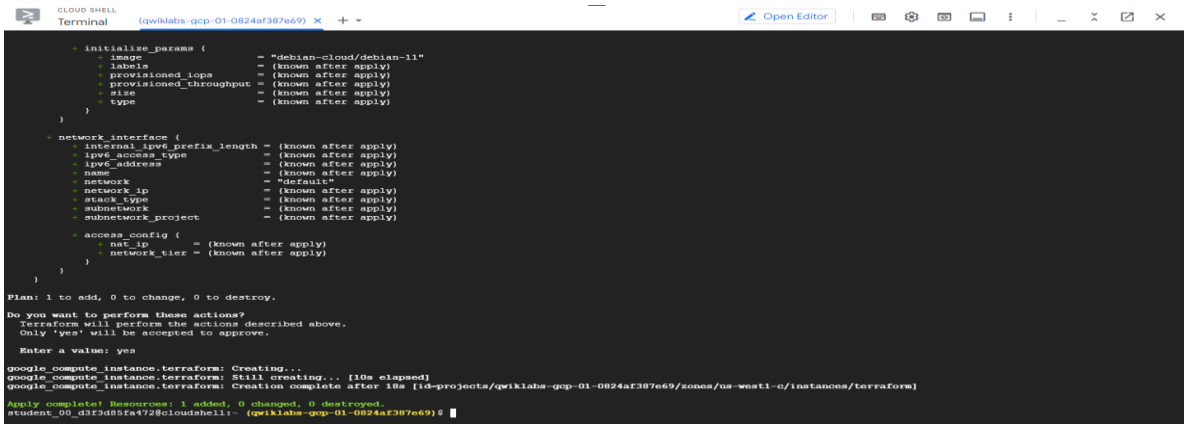


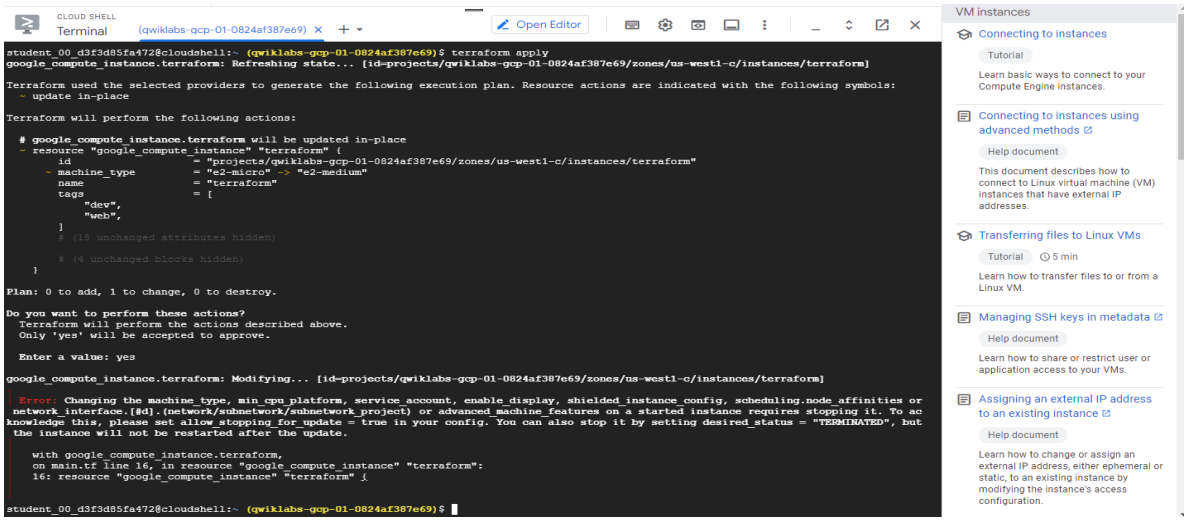
Étape 2 : Infrastructure as Code avec Terraform

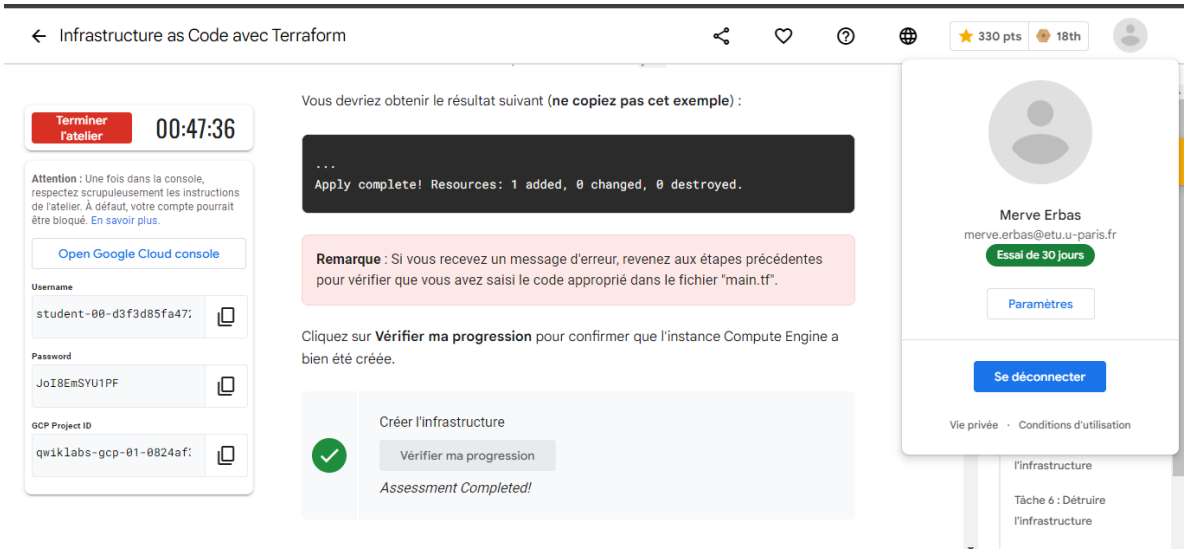
- Vérifier l'installation de Terraform et ajouter un fournisseur Google Cloud



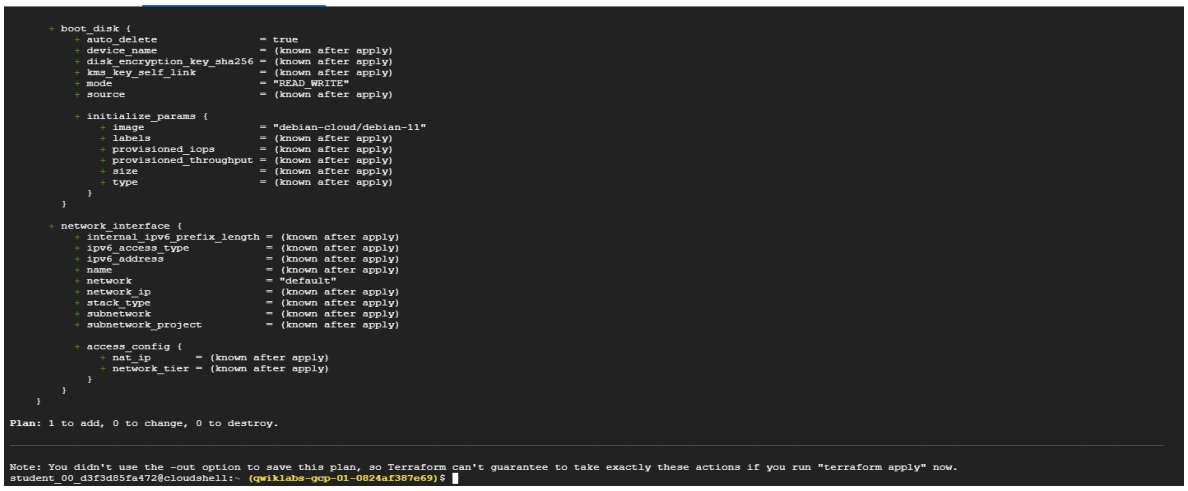
- **Créer l'infrastructure**

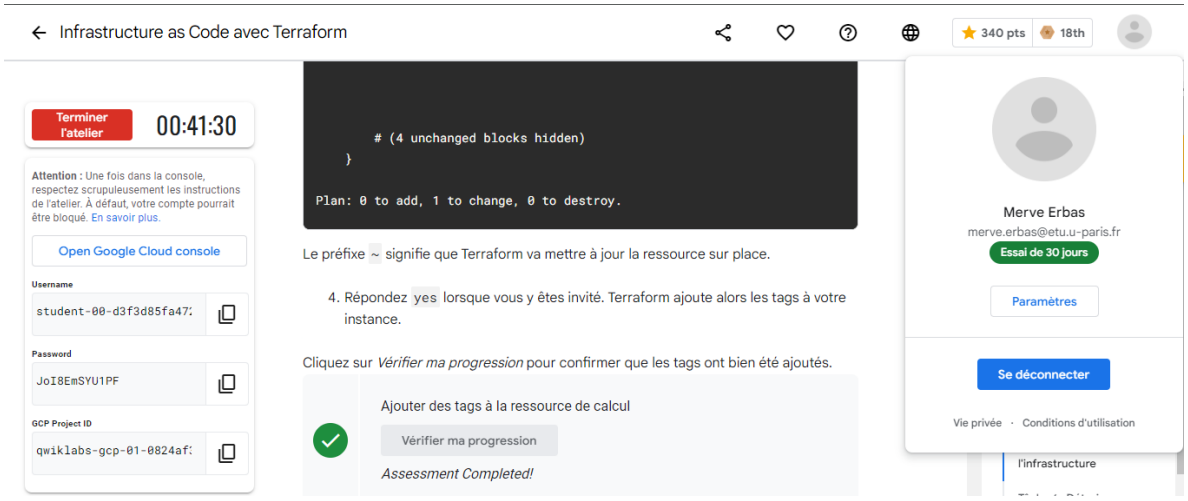




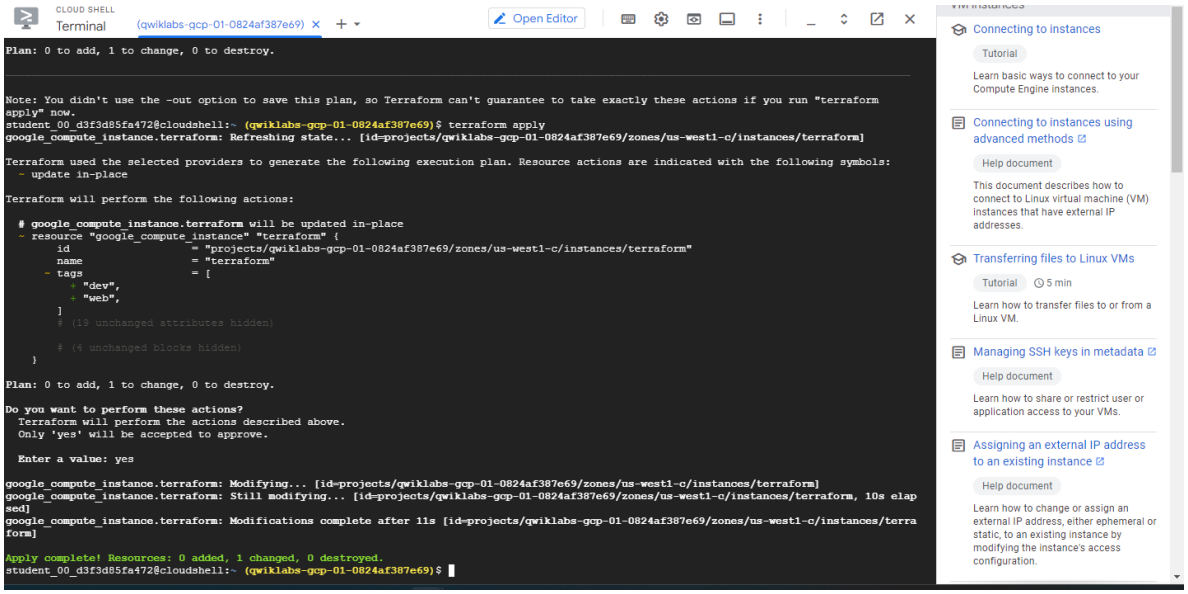


- Modifier l'infrastructure



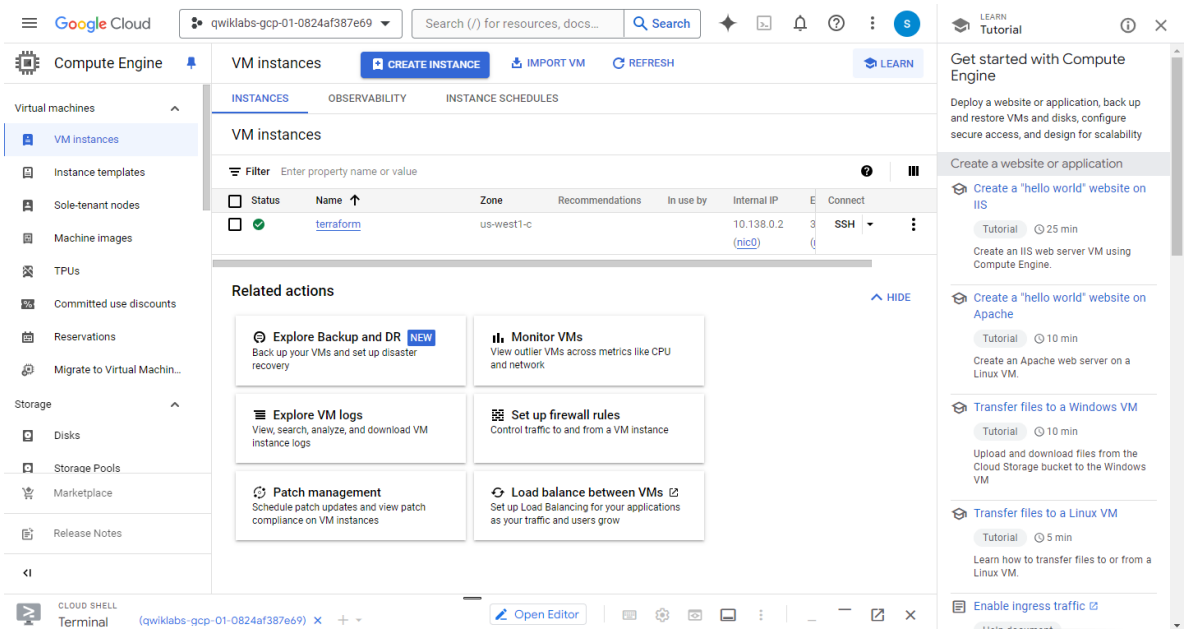


- Changer le type de machine sans arrêter la VM

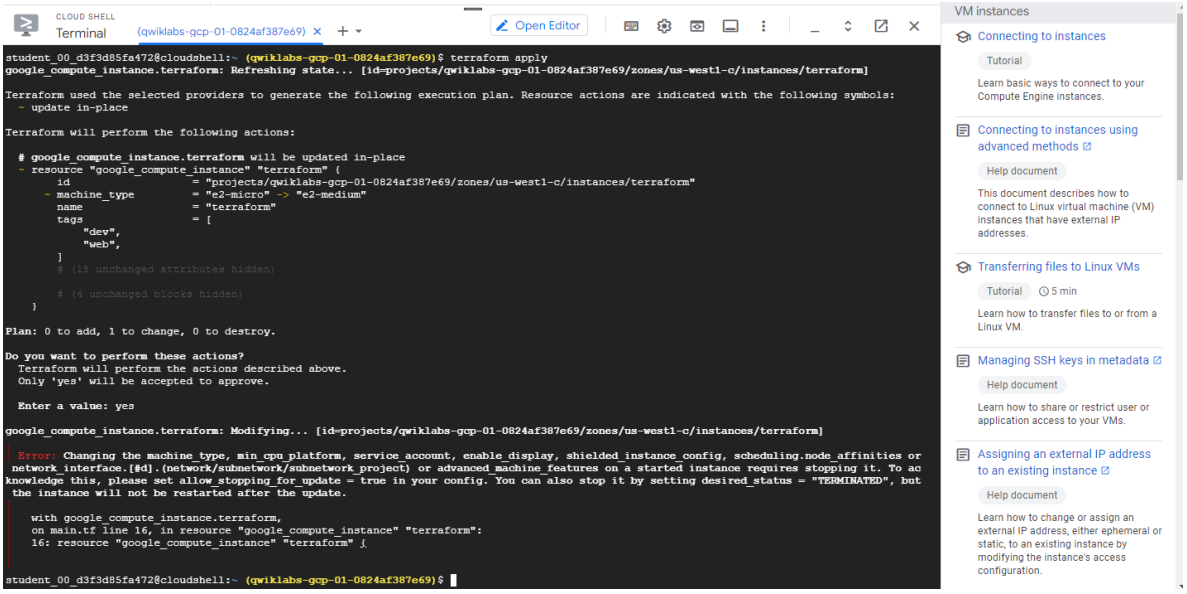


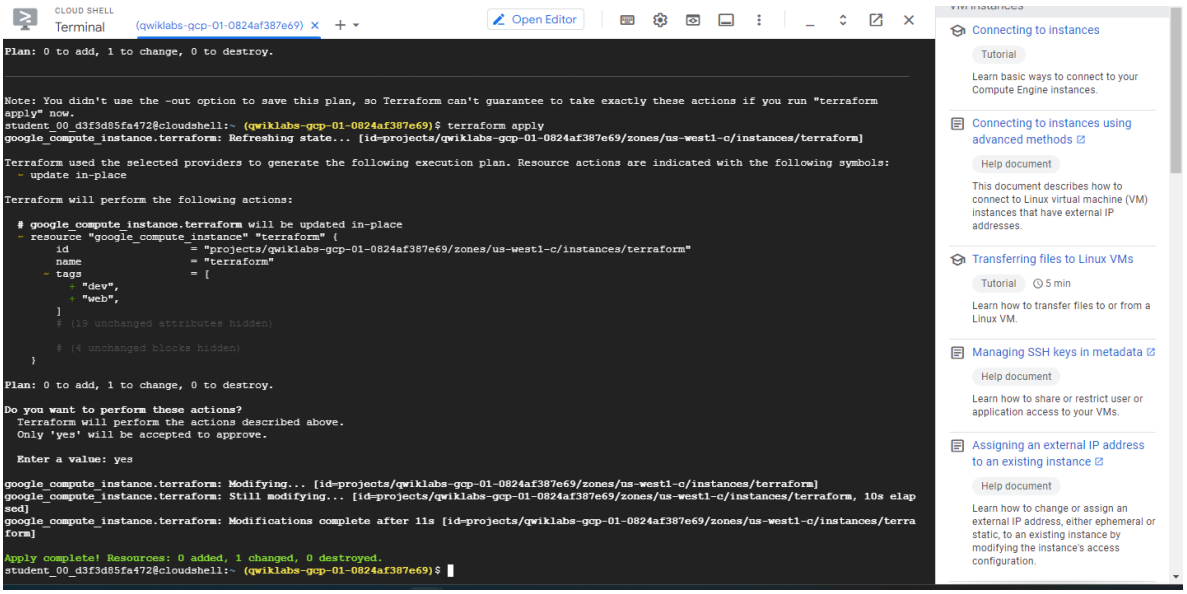
-

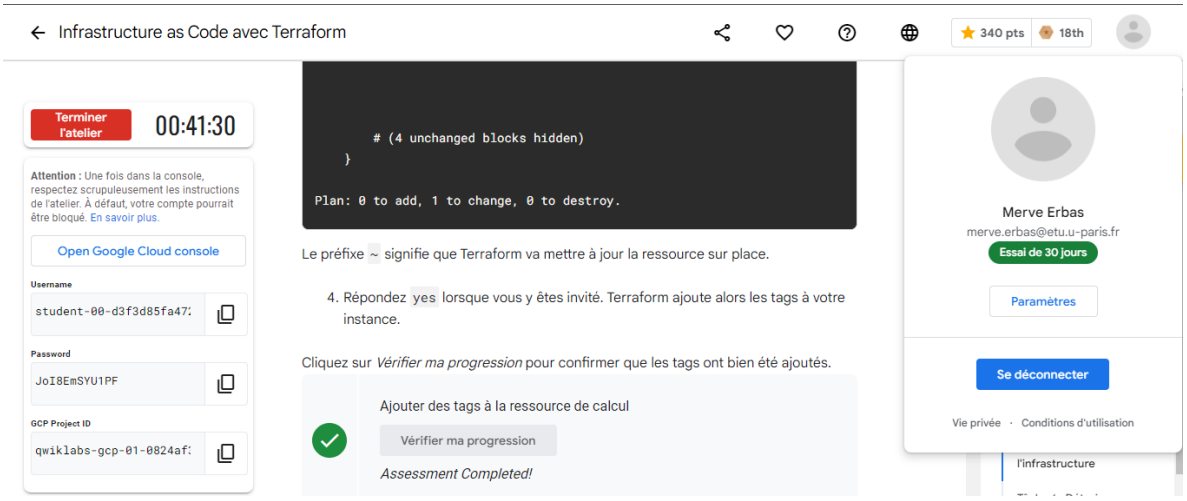
- **Vérifier le résultat dans la console Cloud**



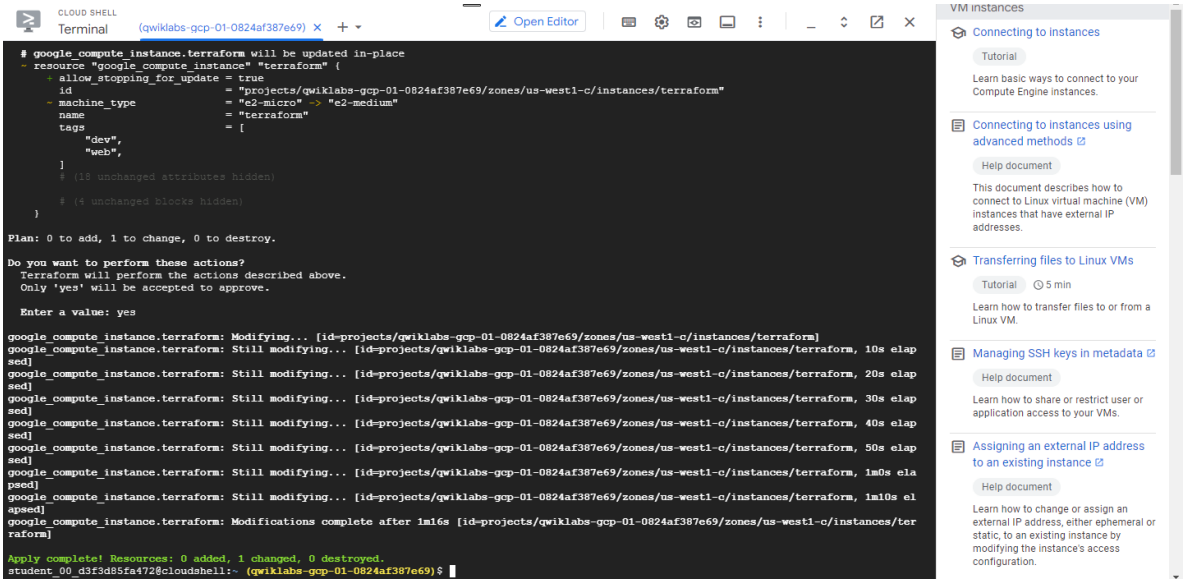
-Modifier insfracture

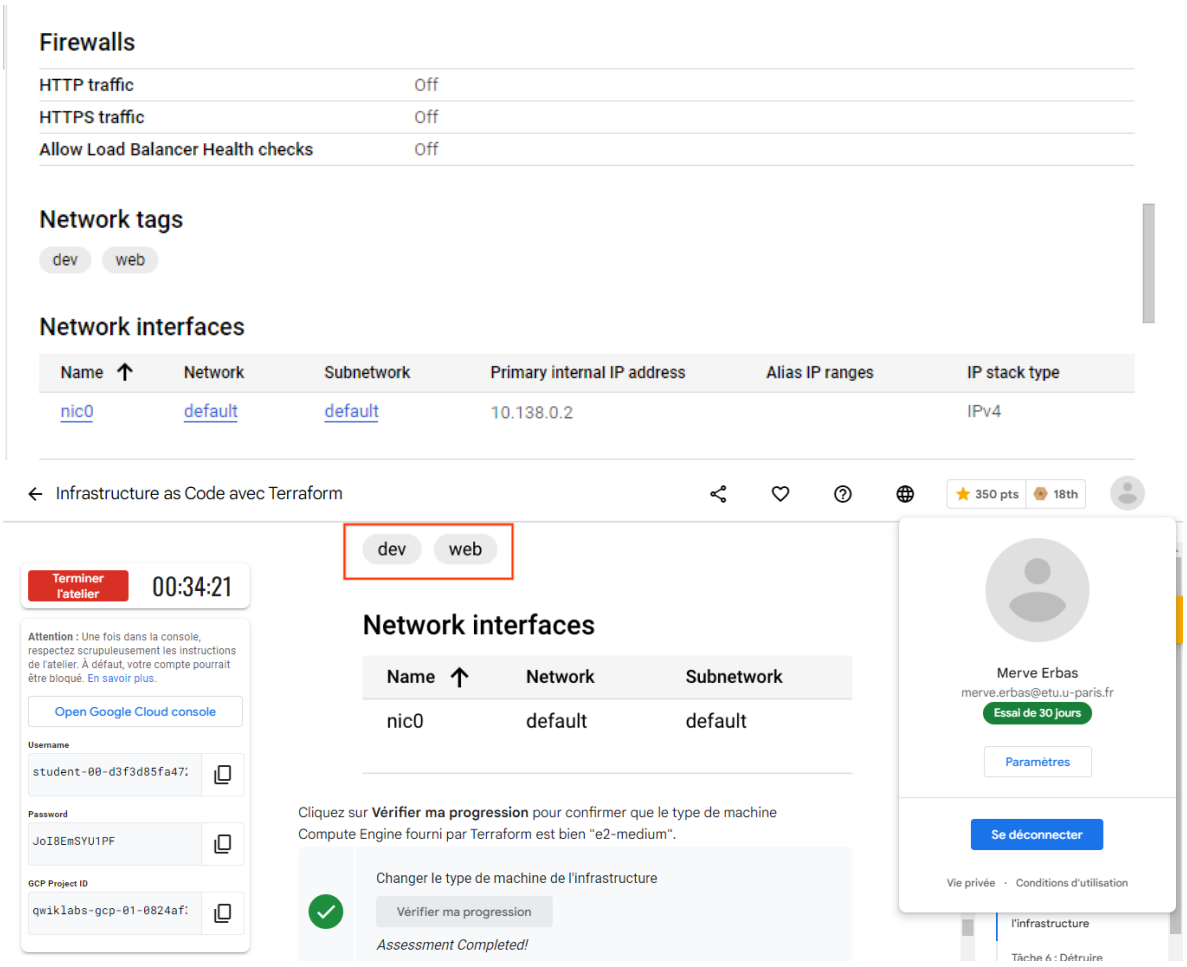

CLOUD SHELL
Terminal
(qwiklabs-gcp-01-0824af387e69)
Open Editor
student_00_d3f3d85fa472@cloudshell:~ (qwiklabs-gcp-01-0824af387e69)$ terraform apply
google_compute_instance.terraform: Refreshing state... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
~ update in-place
Terraform will perform the following actions:
# google_compute_instance.terraform will be updated in-place
~ resource "google_compute_instance" "terraform" {
id = "projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform"
~ machine_type = "e2-micro" -> "e2-medium"
name = "terraform"
tags = [
"dev",
"web",
]
# (18 unchanged attributes hidden)
# (4 unchanged blocks hidden)
}
Plan: 0 to add, 1 to change, 0 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
google_compute_instance.terraform: Modifying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
Error: Changing the machine_type, min_cpu_platform, service_account, enable_display, shielded_instance_config, scheduling.node_affinities or network_interface.[#d].(network/subnetwork/subnetwork_project) or advanced_machine_features on a started instance requires stopping it. To acknowledge this, please set allow_stopping_for_update = true in your config. You can also stop it by setting desired_status = "TERMINATED", but the instance will not be restarted after the update.
with google_compute_instance.terraform,
on main.tf line 16, in resource "google_compute_instance" "terraform":
16: resource "google_compute_instance" "terraform" {
student_00_d3f3d85fa472@cloudshell:~ (qwiklabs-gcp-01-0824af387e69)$
VM instances
Connecting to instances
Tutorial
Learn basic ways to connect to your Compute Engine instances.
Connecting to instances using advanced methods
Help document
This document describes how to connect to Linux virtual machine (VM) instances that have external IP addresses.
Transferring files to Linux VMs
Tutorial
5 min
Learn how to transfer files to or from a Linux VM.
Managing SSH keys in metadata
Help document
Learn how to share or restrict user or application access to your VMs.
Assigning an external IP address to an existing instance
Help document
Learn how to change or assign an external IP address, either ephemeral or static, to an existing instance by modifying the instance's access configuration.


CLOUD SHELL
Terminal
(qwiklabs-gcp-01-0824af387e69)
Open Editor
Plan: 0 to add, 1 to change, 0 to destroy.
Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
student_00_d3f3d85fa472@cloudshell:~ (qwiklabs-gcp-01-0824af387e69)$ terraform apply
google_compute_instance.terraform: Refreshing state... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
~ update in-place
Terraform will perform the following actions:
# google_compute_instance.terraform will be updated in-place
~ resource "google_compute_instance" "terraform" {
id = "projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform"
name = "terraform"
~ tags = [
+ "dev",
+ "web",
]
# (19 unchanged attributes hidden)
# (4 unchanged blocks hidden)
}
Plan: 0 to add, 1 to change, 0 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
google_compute_instance.terraform: Modifying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
google_compute_instance.terraform: Still modifying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 10s elapsed]
google_compute_instance.terraform: Modifications complete after 11s [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
Apply complete! Resources: 0 added, 1 changed, 0 destroyed.
student_00_d3f3d85fa472@cloudshell:~ (qwiklabs-gcp-01-0824af387e69)$
Connecting to instances
Tutorial
Learn basic ways to connect to your Compute Engine instances.
Connecting to instances using advanced methods
Help document
This document describes how to connect to Linux virtual machine (VM) instances that have external IP addresses.
Transferring files to Linux VMs
Tutorial
5 min
Learn how to transfer files to or from a Linux VM.
Managing SSH keys in metadata
Help document
Learn how to share or restrict user or application access to your VMs.
Assigning an external IP address to an existing instance
Help document
Learn how to change or assign an external IP address, either ephemeral or static, to an existing instance by modifying the instance's access configuration.



- Changer le type de machine sans arrêter la VM




Firewalls
HTTP traffic Off
HTTPS traffic Off
Allow Load Balancer Health checks Off
Network tags
dev web
Network interfaces
Name Network Subnetwork Primary internal IP address Alias IP ranges IP stack type
nic0 default default 10.138.0.2 IPv4
Infrastructure as Code avec Terraform
350 pts 18th
Terminer l'atelier 00:34:21
Attention : Une fois dans la console, respectez scrupuleusement les instructions de l'atelier. À défaut, votre compte pourrait être bloqué. En savoir plus.
Open Google Cloud console
Username
student-00-d3f3d85fa47;
Password
JoI8EmSYU1PF
GCP Project ID
qwiklabs-gcp-01-0824af;
dev web
Network interfaces
Name Network Subnetwork
nic0 default default
Cliquez sur Vérifier ma progression pour confirmer que le type de machine Compute Engine fourni par Terraform est bien "e2-medium".
Changer le type de machine de l'infrastructure
Vérifier ma progression
Assessment Completed!
Merve Erbas
merve.erbas@etu.u-paris.fr
Essai de 30 jours
Paramètres
Se déconnecter
Vie privée · Conditions d'utilisation
l'infrastructure
Tâche 6 : Détruire

Machine configuration

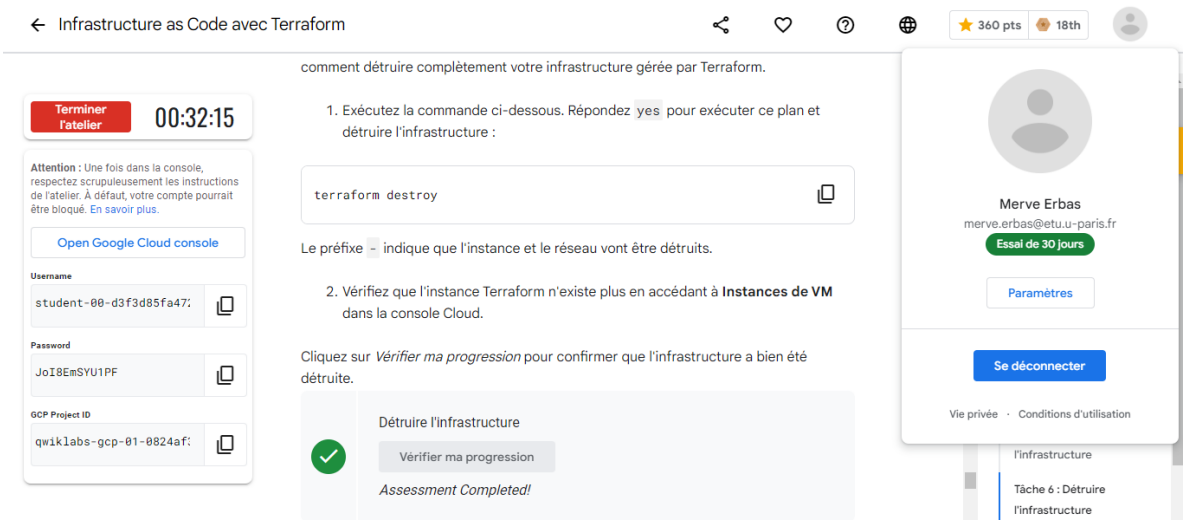
Machine type	e2-medium
CPU platform	Intel Broadwell
Minimum CPU platform	None
Architecture	x86/64
vCPUs to core ratio	–
Custom visible cores	–
Display device	Disabled Enable to use screen capturing and recording tools
GPUs	None
Resource policies	

Networking

Public DNS PTR Record	None
Total egress bandwidth tier	–
NIC type	–

- Détruire l'infrastructure

```
          - automatic_restart   = true -> null
          - min_node_cpus       = 0 -> null
          - on_host_maintenance = "MIGRATE" -> null
          - preemptible         = false -> null
          - provisioning_model  = "STANDARD" -> null
        }

      - shielded_instance_config {
          - enable_integrity_monitoring = true -> null
          - enable_secure_boot          = false -> null
          - enable_vtpm                 = true -> null
        }
    }

Plan: 0 to add, 0 to change, 1 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

google_compute_instance.terraform: Destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform]
google_compute_instance.terraform: Still destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 10s ela
psed]
google_compute_instance.terraform: Still destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 20s ela
psed]
google_compute_instance.terraform: Still destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 30s ela
psed]
google_compute_instance.terraform: Still destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 40s ela
psed]
google_compute_instance.terraform: Still destroying... [id=projects/qwiklabs-gcp-01-0824af387e69/zones/us-west1-c/instances/terraform, 50s ela
psed]
google_compute_instance.terraform: Destruction complete after 53s

Destroy complete! Resources: 1 destroyed.
student_00_d3f3d85fa472@cloudshell:~ (qwiklabs-gcp-01-0824af387e69)$
```



4. Conclusion

Ces ateliers ont permis de comprendre les principes de base de Google Cloud Platform, de déployer des applications, d'utiliser des services spécifiques de GCP, de configurer et sécuriser les services cloud, et d'analyser des données en utilisant les outils de Google Cloud. Les compétences acquises sont essentielles pour le développement, la gestion et la mise à l'échelle des applications cloud modernes.

Pour plus de détails et pour consulter les sorties des commandes et les captures d'écran, veuillez visiter mon GitHub.